



UNIVERSITI PUTRA MALAYSIA

**IMPROVED MULTICROSSOVER GENETIC ALGORITHM FOR TWO-
DIMENSIONAL RECTANGULAR BIN PACKING PROBLEM**

**MARYAM SARABIAN
FS 2010 05**





IMPROVED MULTICROSSOVER GENETIC ALGORITHM FOR TWO-DIMENSIONAL RECTANGULAR BIN PACKING PROBLEM

MARYAM SARABIAN

**MASTER OF SCIENCE
UNIVERSITI PUTRA MALAYSIA**

2010



To all who supported me during this research.

Abstract of the thesis presented to the Senate of Universiti Putra Malaysia
in fulfilment of requirements for the degree of Master of Science

**IMPROVED MULTICROSSOVER GENETIC ALGORITHM FOR TWO-
DIMENSIONAL RECTANGULAR BIN PACKING PROBLEM**

By

MARYAM SARABIAN

March 2010

Chairman: Lee Lai Soon, PhD

Faculty: Science

Bin Packing Problem is a branch of Cutting and Packing problems which has many applications in wood and metal industries. In this research we focus on non-oriented case of Two-Dimensional Rectangular Bin Packing Problem (2DRBPP). The objective of this problem is to pack a given set of small rectangles, which may be rotated by 90° , without overlaps into a minimum numbers of identical large rectangles.

Our aim is to improve the performance of the MultiCrossover Genetic Algorithm (MXGA) proposed from the literature for solving the problem. We focus on four major components of the MXGA which consist of selection, crossover, mutation and replacement. Initial computational experiments are conducted independently on the named components using some benchmark problem instances. The most competitive techniques from each component are combined to form a new algorithm called

Improved MXGA (MXGA i). Extensive computational experiments are performed using benchmark data sets to assess the effectiveness of the proposed algorithm. The MXGA i is shown to be competitive when compared with MXGA, Standard GA, Unified Tabu Search (UTS) and Randomised Descent Method (RDM).

Abstrak tesis yang dikemukakan kepada Senat Universiti Putra Malaysia
sebagai memenuhi keperluan untuk ijazah Master Sains

**PENAMBAHBAIKAN ALGORITMA GENETIK MULTI-LINTASAN BAGI
PEMBUNGKUSAN BEKAS SEGIEMPAT TEPAT BERDIMENSI DUA**

Oleh

MARYAM SARABIAN

Mac 2010

Pengerusi: Lee Lai Soon, PhD

Fakulti: Sains

Masalah Pembungkusan Bekas merupakan antara masalah dalam Pemotongan dan Pembungkusan yang mana banyak diaplikasikan dalam industri kayu dan logam. Dalam kajian ini, tumpuan adalah kepada kes yang bersifat bukan orientasi bagi Masalah Pembungkusan Segiempat Tepat Dua-Dimensi (2DRBPP). Objektif yang ingin dicapai ialah untuk membungkus set segiempat tepat kecil yang boleh diputarakan 90° tanpa berlaku pertindihan di dalam segiempat tepat serupa dengan jumlah minimum.

Matlamat kami adalah untuk memperbaiki prestasi Algoritma Genetik Multi-Lintasan (MXGA) yang telah dicadang dari kesusasteraan bagi menyelesaikan masalah yang dihadapi. Terdapat empat komponen utama MXGA yang diberikan perhatian iaitu pemilihan, lintasan, mutasi dan penggantian. Pada peringkat awal, eksperimen komputasi telah dilakukan terhadap komponen-komponen yang disenaraikan dengan

menggunakan contoh masalah sebagai tanda aras. Teknik yang paling kompetitif daripada setiap komponen dipilih dan digabungkan bagi membentuk satu algoritma baharu yang dikenali sebagai Penambahbaikan Algoritma Genetik Multi-Lintasan (MXGA_i). Seterusnya, eksperimen berkomputer lanjutan telah dijalankan dengan menggunakan data mengikut tanda aras yang ditetapkan. Ia bertujuan untuk mengenal pasti keberkesanan algoritma yang telah dicadangkan. Hasil kajian menunjukkan MXGA_i adalah lebih kompetitif berbanding MXGA, GA piawai, Carian Tabu Seragam (UTS) dan Kaedah Rawakan Menurun (RDM).

ACKNOWLEDGEMENTS

Hereby I would like to express my gratitude to all those who gave me the possibility to complete this thesis. I am deeply indebted to my supervisor Dr. Lee Lai Soon for his supports, valuable suggestions and encouragements during my research and writing of this thesis. I want to thank the Ministry of Science, Technology and Innovation (MOSTI) for funding this project under the grant number of 06-01-04-SF0085.

Furthermore I want to express my thanks to the supervisory committee members Assoc. Prof. Dr. Mohd Rizam Abu Bakar and Assoc. Prof. Dr. Habshah Midi for their supports.

Last but not least, I would like to give a special thanks to my lovely husband, my dear family and friends for their patience and love, which helped me to complete this work.

I certify that a Thesis Examination Committee has met on 25 March 2010 to conduct the final examination of Maryam Sarabian on her thesis entitled “IMPROVED MULTICROSSOVER GENETIC ALGORITHM FOR TWO-DIMENSIONAL RECTANGULAR BIN PACKING PROBLEM” in accordance with the Universities and University Colleges Act 1971 and the Constitution of the Universiti Putra Malaysia [P.U.(A) 106] 15 March 1998. The Committee recommends that the student be awarded the Master of Science degree.

Members of the Thesis Examination Committee were as follows:

Zanariah Abdul Majid, PhD

Lecturer
Faculty of Science
Universiti Putra Malaysia
(Chairperson)

Leong Wah June, PhD

Lecturer
Faculty of Science
Universiti Putra Malaysia
(Internal Examiner)

Norihan Md. Arifin, PhD

Associate Professor
Faculty of Science
Universiti Putra Malaysia
(Internal Examiner)

Hj. Mustafa Mamat, PhD

Lecturer
Faculty of Science and Technology
Universiti Malaysia Terengganu
Malaysia
(External Examiner)

BUJANG BIN KIM HUAT, PhD

Professor and Deputy Dean

School of Graduate Studies

Universiti Putra Malaysia

Date:

This Thesis was submitted to the Senate of Universiti Putra Malaysia and has been accepted as a fulfilment of the requirement for the degree of Master of Science. The members of Supervisory Committee were as follows:

Lee Lai Soon, PhD

Lecturer

Faculty of Science

Universiti Putra Malaysia

(Chairman)

Mohd Rizam Abu Bakar, PhD

Associate Professor

Faculty of Science

Universiti Putra Malaysia

(Member)

Habshah Midi, PhD

Associate Professor

Faculty of Science

Universiti Putra Malaysia

(Member)

HASANAH MOHD GHAZALI, PhD

Professor and Dean

School of Graduate Studies

Universiti Putra Malaysia

Date: 13 May 2010

DECLARATION

I declare that this thesis is my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously, and is not concurrently, submitted for any other degree at Universiti Putra Malaysia or at any other institution.

MARYAM SARABIAN

Date:

TABLE OF CONTENTS

	Page
DEDICATION	ii
ABSTRACT	iii
ASTRAKT	v
ACKNOWLEDGMENTS	vii
APPROVAL	viii
DECLARATION	x
LIST OF TABLES	xiv
LIST OF FIGURES	xvi
 CHAPTER	
1 INTRODUCTION	1
1.1 Scope of Study	2
1.2 Problem Statement	4
1.3 Objectives	4
1.4 Overview of Thesis	5
 2 HEURISTIC AND METAHEURISTIC APPROACHES	6
2.1 Introduction	6
2.2 Complexity Theory	6
2.3 Cutting and Packing (C&P) Problems	9
2.3.1 Wäscher <i>et al.</i> 's Typology	9
2.3.2 Basic Types of C&P Problems	10
2.3.3 Intermediate Problem Types (IPT)	12
2.4 Heuristic and Metaheuristic Approaches for 2DRBPP	14
2.4.1 Heuristic Approaches	15
2.4.2 Metaheuristic Approaches	20
2.4.3 Application of Metaheuristics for Solving 2DRBPP	26
2.5 Summary	31
 3 GENETIC ALGORITHM	32
3.1 Introduction	32
3.2 Representation	32

3.3	Initial Population	34
3.4	Fitness Function	35
3.5	Selection Mechanism	36
3.6	Crossover Operator	41
3.7	Mutation Operator	50
3.8	Replacement Strategy	54
3.9	Summary	55
4	IMPROVED MULTICROSSOVER GENETIC ALGORITHM (MXGA_i)	57
4.1	Introduction	57
4.2	Implementation of MXGA _i	58
4.2.1	Representation	58
4.2.2	Initial Population	59
4.2.3	Decoding Stage	59
4.2.4	Fitness Function	62
4.2.5	Selection Mechanism	64
4.2.6	Crossover Operator	65
4.2.7	Swap Operator	75
4.2.8	Mutation Operator	75
4.2.9	Replacement Strategy	76
4.3	Other Local Search methods	77
4.3.1	Unified Tabu Search	77
4.3.2	Randomised Descent Method (RDM)	82
4.4	Experimental Design	84
4.5	Summary	86
5	RESULTS AND DISCUSSION	88
5.1	Introduction	88
5.2	Initial Investigations of MXGA	88
5.2.1	Comparison of Different Selection Mechanisms in MXGA	89
5.2.2	Comparison of Different Crossover Operators in MXGA	92
5.2.3	Comparison of Different Mutation Operators in MXGA	95
5.2.4	Comparison of Different Replacement Strategies in MXGA	97

5.3	Comparison of MXGA <i>i</i> with MXGA and other Local Search Algorithms	98
5.4	Conclusions	104
6	CONCLUSIONS AND FUTURE WORKS	105
6.1	Introduction	105
6.2	Conclusions	105
6.3	Future Works	107
	REFERENCES	108
	APPENDIX A	115
	BIODATA OF STUDENT	127
	LIST OF PUBLICATIONS	128

LIST OF TABLES

Table	Page
2.1: Wäscher <i>et al.</i> 's Typology	10
2.2: Landscape of IPT: Output Maximisation	13
2.3: Landscape of IPT: Input Minimisation	13
4.1: Classes for the problem instances	85
5.1: Comparison of different Selection Mechanisms in MXGA	90
5.2: Comparison of different Crossover Operators in MXGA	93
5.3: Comparison of different Mutation Operators in MXGA	95
5.4: Comparison of different Replacement Strategies in MXGA	97
5.5: Implementation of generic design variables for MXGA _i , MXGA and SGA	100
5.6: Implementation of generic design variables for UTS and RDM	100
5.7: A Comparison of MXGA _i with MXGA, SGA, UTS and RDM	102
A.1: Comparison of different Selection Mechanisms in MXGA for $n = 60, 80$ and 100 of class V	115
A.2: Comparison of different Selection Mechanisms in MXGA for $n = 60, 80$ and 100 of class VII	116
A.3: Comparison of different Selection Mechanisms in MXGA for $n = 60, 80$ and 100 of class VIII	117
A.4: Comparison of different Crossover Operators in MXGA for $n = 60, 80$ and 100 of class V	118
A.5: Comparison of different Crossover Operators in MXGA for $n = 60, 80$ and 100 of class VII	119
A.6: Comparison of different Crossover Operators in MXGA for $n = 60, 80$ and 100 of class VIII	120
A.7: Comparison of different Mutation Operators in MXGA for $n = 60, 80$ and 100 of class V	121
A.8: Comparison of different Mutation Operators in MXGA for $n = 60, 80$ and 100 of class VII	122

A.9: Comparison of different Mutation Operators in MXGA for $n = 60, 80$ and 100 of class VIII	123
A.10: Comparison of different Replacement Strategies in MXGA for $n = 60, 80$ and 100 of class V	124
A.11: Comparison of different Replacement Strategies in MXGA for $n = 60, 80$ and 100 of class VII	125
A.12: Comparison of different Replacement Strategies in MXGA for $n = 60, 80$ and 100 of class VIII	126

LIST OF FIGURES

Figure	Page
2.1: A simple diagram of P and NP	9
2.2: Basic problem types of C&P	11
2.3: A pseudocode for Descent Method	22
2.4: A pseudocode for Tabu Search	23
2.5: Genetic Algorithm's Structure	25
3.1: An example for binary representation	33
3.2: An example for permutation representation	33
3.3: An example for value representation	33
3.4: An example for matrix representation	34
3.5: An example for 1-Point Crossover	42
3.6: An example for 2-Point Crossover	42
3.7: An example for Uniform Crossover	43
3.8: An example for Partially Mapped Crossover	44
3.9: An example for Order Crossover	45
3.10: An example for 1X Crossover	45
3.11: An example for Cycle Crossover	46
3.12 : An example for Maximal Preservative Crossover	47
3.13: An example for Order Based Crossover	47
3.14: An example for Position Based Crossover	48
4.1: An example of an individual in MXGAi	59
4.2: Implementation of the <i>repack</i> strategy	61
4.3: An example for PMX _m in MXGAi	67
4.4: An example for the proposed 1X _m	70

4.5: An example for the proposed OX_m	71
4.6: An example for the proposed MPX_m	72
4.7: An example for the proposed MMX_m	73
4.8: Possible offspring after applying the <i>repair mechanism</i>	74
4.9: General structure of UTS	79
4.10: SEARCH procedure in UTS	80
4.11: DIVERSIFICATION procedure in UTS	82
4.12: SEARCH procedure in RDM	83

CHAPTER 1

INTRODUCTION

Cutting and Packing (C&P) problems are classified as combinatorial optimization problems. These types of problems consist of two sets of elements, namely

- a set of large objects (input, supply), and
- a set of small items (output, demand)

The objective of these problems is minimizing the overall size of unused part of the large objects or maximizing the number of small items to be packed in the large objects. These types of problems have many applications in business and industry (e.g. wood, glass and textile industries, vehicle or container loading, newspaper paging and etc).

C&P problems can be defined in one, two, three or larger number (n) of dimensions and a solution of the problem may result in applying some or all large objects, and some or all small items. Bin Packing Problem (BPP) is a type of C&P problems which characterised by assortment of all small items into minimum number of large objects. This problem has many applications in wood and glass industries (cutting the rectangular component from large sheets of material) and in newspapers paging (arrangement of articles and advertisements into pages). BPP is classified as a class of NP-hard problem by Garey and Johnson [25].

1.1 Scope of Study

In this study we will concentrate on non-oriented case of Two-Dimensional Rectangular Single Bin Size Bin Packing Problem (2DRSBSBPP) based on classification of Wäscher *et al.* [61]. Without loss of generality, the problem will be referred as Two-Dimensional Rectangular Bin Packing Problem (2DRBPP) henceforth. In this problem a given set of two-dimensional differently sized small rectangles (items), which may be rotated by 90° , has to be packed without being overlapped into the minimum number of identical large objects (bins). It is worth noting that the additional requirements for the 2DRBPP in this study are as below:

1. All the rectangles are packed in *non-guillotine cuts pattern*: items are not obtained from a sequence of edge-to-edge cuts.
2. All the rectangles are packed in an *orthogonal packing pattern*: the edges of the rectangles are parallel to the edges of the bins.

Since the 2DRBPP is a NP-hard problem, exact algorithms are only able to solve small to medium size problem instances. Big size problem instances with large number of rectangles have to be solved by heuristic or local search methods. This research concentrates on local search methods as a tool for solving the problem.

Genetic Algorithm is an adaptive local search method which was first invented by Holland [33]. This algorithm is based on the genetic process of biological organisms.

According to the Darwin's principle "survival of the fittest", the organisms which are most capable of acquiring resources and attracting mates will generate more offspring. By abstracting the evolutionary principles to a real world problem, GA is able to find an optimal solution.

An implementation of Holland's GA begins with a random population of individuals. Each individual represents a feasible solution to the problem and is composed of a string of genes with the defined length. In each generation, the individuals are selected from the population according to their fitness values in order to generate new offspring via crossover operator. In the case that the crossover is not applied to the selected individuals, the offspring will be generated by the exact duplication of the parents. After performing the crossover operator, mutation will take place. At the end of each generation the parent population will be replaced by the offspring population by means of the replacement strategy. The process will be repeated for a fixed number of generations or a fixed amount of time with the hope of finding the optimal solution.

MultiCrossover Genetic Algorithm (MXGA) is a specific variant of GA which proposed by Lee [40]. In the MXGA, offspring for the next generation are selected from a list of temporary offspring generated via a multicrossover operator.

1.2 Problem Statement

The vast majority of the literatures concern heuristics and local search methods for solving 2DRBPP. Although computational results in the literature indicate that MXGA achieved better quality solutions compared to Standard Genetic Algorithm (SGA) but there are still rooms for improving the MXGA.

Since the crossover operators which are applied in the multicrossover process of MXGA (Lee [40]), are standard 1-Point and 2-Point crossover operators it is predicted that applying the other crossover operators in the MXGA can improve the quality of solutions. The improvement can also be done by changing the other main components of MXGA such as selection mechanism, mutation operator and replacement strategy.

1.3 Objectives

Generally the objectives of this study are as below:

1. Improving the implementation of MXGA for solving the problem. This can be done by focusing on four major components of the MXGA namely selection mechanism, crossover operator, mutation operator and replacement strategy. Our new proposed algorithm is constructed by combining the most competitive techniques from each component.

2. Comparing the effectiveness of the new proposed algorithm with MXGA, SGA and other local search methods such as Unified Tabu Search (UTS) and Randomised Descent Method (RDM). We hope that our new proposed algorithm will be able to achieve a better quality solutions compared to other named local search methods.

1.4 Overview of Thesis

The remainder of this thesis is structured as follows: Chapter 2 begins with introducing the concept of time complexity and follows by giving a general overview of C&P problems. Different heuristic and metaheuristic approaches for solving 2DRBPP are also presented in Chapter 2. Detailed descriptions of the main components of GA and some of the well-known approaches in each component are given in Chapter 3. Different components of our proposed algorithm, the implementation of the other local search methods which are applied for solving the problem in this study and the experimental design are described in Chapter 4.

Initial investigations on the four major components of MXGA are given in Chapter 5, also a comparison is made between our new proposed algorithm, MXGA, SGA, UTS and RDM through extended experimental results using benchmark data sets. We give conclusions and describe possible future works in Chapter 6.

CHAPTER 2

HEURISTIC AND METAHEURISTIC APPROACHES

2.1 Introduction

This chapter is structured as follows: in Section 2.2 we review the concept of time complexity. Definition of C&P problems and Wäscher's typology are given in Section 2.3. Section 2.4 starts with giving a definition for heuristic and metaheuristic and follows by discussing some of the well-known heuristic and metaheuristic approaches for solving 2DRBPP. A summary of this chapter is given in Section 2.5.

2.2 Complexity Theory

Computational complexity measures how much time is needed to solve different problems. This will help to find out whether a problem is easy or hard. If the problem is easy it can be solved as a linear program or network model. It is not easy to find an exact solution for the hard problems. In this case the problem needs to be solved by heuristics or local search algorithms. In this section we concentrate on the time complexity theory. The definitions in this section are extracted from Tovey [58] and Whitley and Watson [62].

The time complexity of a problem is the number of steps that it takes to solve an instance of the problem as a function of the size of the input length (usually measured in bits) using the most efficient algorithm. For example, consider an instance that is n bits long which can be solved in n^3 steps, so in this case the problem has a time complexity of n^3 . Big- O notation is generally applied to interpret the time complexity of a problem. If a problem's time complexity is $O(n^2)$ on one typical computer, then it will also have time complexity of order $O(n^2)$ on most other computers, so this notation allows us to generalize away from the details of a particular computer.

Suppose an algorithm solves a problem of size n in at most $12n^3 + 8n^2 + 15$ steps. For such functions, we are primarily interested in the rate of growth as n increases. Therefore, the difference between $12n^3$ and n^3 is not really important. We also can ignore the lower order terms, because at the large sizes it is the highest degree that determines the rate of growth. So we say that the algorithm is of order $O(n^3)$, it means this algorithm requires $O(n^3)$ time. This symbolism is a reminder that this function expresses the worst case behaviour at sufficiently large sizes.

Such algorithms with running times of orders $O(\log n)$, $O(n \log n)$, $O(n)$, $O(n^2)$, $O(n^3)$ are called 'polynomial-time' algorithms. Algorithms with complexities which cannot be bounded by polynomial functions are called 'exponential-time' algorithms. In practice exponential time algorithms are slower than polynomial time algorithms.